



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

am

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/977,513	10/12/2001	David S. Allison	0007056-0198/P5941	3871
32615	7590	05/06/2005	EXAMINER	
OSHA LIANG L.L.P./SUN 1221 MCKINNEY, SUITE 2800 HOUSTON, TX 77010			VU, TUAN A	
			ART UNIT	PAPER NUMBER
			2193	
DATE MAILED: 05/06/2005				

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/977,513

Applicant(s)

ALLISON, DAVID S.

Examiner

Tuan A. Vu

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
 - If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
 - If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
 - Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 03 December 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-30 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-30 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to the Applicant's response filed 12/03/2004.

As indicated in Applicant's response, claims 1, 6, 11, 16, 21, and 26 have been amended.

Claims 1-30 are pending in the office action.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

3. Claims 1, 6, 11, 16, 21, and 26 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claims 1, 6, 11, 16, 21, and 26 rejected under 35 U.S.C. 112, second paragraph, as being incomplete for omitting essential steps, such omission amounting to a gap between the steps. See MPEP § 2172.01. The omitted steps are: some steps accomplishing or putting forth a dynamically typed object definition in conjunction with retrieving a type or a value when the corresponding operator is executed; because executing an operator (size or type) as recited does not explicitly amount to defining an object using a dynamically-typed language; and vice versa defining an object with dynamically typed language is not equivalent to executing such operator. Such omission is interpreted as if there is no connectivity between executing the operator and the dynamic type definition.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

Art Unit: 2193

A person shall be entitled to a patent unless—

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1-5, 11-15, and 21-25 are rejected under 35 U.S.C. 102(b) as being anticipated by University of Oregon, "TAU Portable Profiling Package", <http://www.acl.lanl.gov/tau>, copyright 1997 (hereinafter OregonU, pp. 1-4); and admitted prior art by (hereinafter APA – see instant specifications: BACKGROUND ART).

As per claim 1, OregonU and APA (*sizeof* - pg. 2, bottom para) disclose a method for determining size value associated with an object at runtime comprising:

executing a computer program; associating a size value with an object during execution of said program; and

retrieving said size value when a size operator is executed (Note: the Examiner will interpret this 'is executed' limitation as if the instruction underlying the resolved operator is being executed or the instruction operating on what represents such operator at runtime is executed for yielding a runtime result) for said object during execution of said program (see APA; see OregonU: e.g. *int func_templ (T x) ... return sizeof(x)* , pg. 2 – Note: a size associated with x is being retrieved at runtime when the instruction underlying the source symbol operator *sizeof* is being executed);

wherein said object is defined using a dynamically-type language (Note: C and C++ support of dynamic type defining via casting, an this well-known feature in C/C++ language reads on the claimed limitation).

As per claims 2 and 3, official notice is taken in programming language using *sizeof* as operator (C/C++), the arguments such as a string or vector or array being passed to such

Art Unit: 2193

operator are known concepts in computer programming languages. Hence, these limitations are implicitly disclosed by APA or OregonU.

As per claim 4, OregonU discloses a value being determined by a function of an object being an instance of a class (OregonU: e.g. *int func_templ (T x) ... return sizeof(x)*, pg. 2)

As per claim 5, OregonU discloses providing an instance of size operator for said class and calling said instance of said size operator (see *class T* and *sizeof(x)* from OregonU pg. 2 – re claim 4).

As per claim 11, OregonU and APA (*sizeof* - pg. 2, bottom para) disclose a runtime size information determiner comprising:

an associating unit configured to associate a size value with an object during execution of a computer program; and

a retrieval unit configured to retrieving said size value when a size operator is executed for said object during execution of said program(see APA; see OregonU: e.g. *int func_templ (T x) ... return sizeof(x)* , pg. 2 – Note: a size associated with x is being retrieved at runtime when the instruction underlying the source symbol operator *sizeof* is being executed);

wherein said object is defined using a dynamically-type language (Note: C and C++ support of dynamic type defining via casting, an this well-known feature in C/C++ language reads on the claimed limitation).

As per claims 12-15, these claims correspond to claims 2-5, respectively; hence are rejected with the corresponding rejections set forth therein.

Art Unit: 2193

As per claim 21, this is a computer program product with readable medium to embody a program code for performing the same step limitations recited in claim 1; hence is rejected with the corresponding rejection as set forth therein.

As per claims 22-25, these claims correspond to claims 2-5, respectively; hence are rejected with the corresponding rejections set forth therein.

6. Claims 6-10, 16-20, and 26-30 rejected under 35 U.S.C. 102(b) as being anticipated by Bruce Eckel, "Thinking in C++ 2nd edition, Volume 2: Standard Libraries & Advanced Topics", 1999, chp. 17, pp. 399-422; url = "citeseer.ist.psu.edu/385233.html" (hereinafter Eckel); and admitted prior art by (alias APA – see instant specifications: BACKGROUND ART).

As per claim 6, APA (*typeof* - pg. 2, bottom para) and Eckel disclose a method for determining type value associated with an object at runtime comprising:

executing a computer program; associating a type value with an object during execution of said program (e.g. *RTTI* - Eckel: pg. 400-404); and

retrieving said type value when a type operator is executed (Note: the Examiner will interpret this 'is executed' limitation as if the instruction underlying the resolved operator is being executed or the instruction operating on what represents such operator at runtime is executed for yielding a runtime result) for said object during execution of said program(see APA; see Eckel: *typeid ()* - pp 404-408);

wherein said object is defined using a dynamically-type language (Note: C and C++ support of dynamic type defining via casting, an this well-known feature in C/C++ language reads on the claimed limitation).

As per claims 7 and 8, official notice is taken in programming language using *typeof* as operator (C/C++ as in APA), the arguments such as a string or an array being passed to such operator are known concepts in computer programming languages. Besides, Eckel discloses type evaluation by getting type information in a structure (e.g. Eckel: pp. 404-410 – Note: a *typeinfo* structure implicitly discloses the object type being retrieved to be either a string or an address – a number- representing such type information structure reference). Hence, these limitations are implicitly disclosed by APA or Eckel.

As per claim 9, based on the object-oriented programming as disclosed by Eckel, the use of one instance of *typeid()* and a method within a class instance is disclosed (re claim 6).

As per claim 10, Eckel discloses providing an instance of type operator for said class and calling said instance of said type operator (see Eckel – re claim 9).

As per claim 16, APA (*typeof* - pg. 2, bottom para) and Eckel disclose a runtime type information determiner comprising:

an associating unit configured to associate a type value with an object during execution of a computer program (e.g. *RTTI* - Eckel: pg. 400-404); and

a retrieval unit configured to retrieving said type value when a type operator is executed for said object during execution of said program(see APA; see Eckel: *typeid ()* - pp 404-408);

wherein said object is defined using a dynamically-type language (Note: C and C++ support of dynamic defining of type via casting, an this well-known feature in C/C++ language reads on the claimed limitation).

As per claims 17-20, these claims correspond to claims 7-10, respectively; hence are rejected with the corresponding rejections set forth therein.

As per claim 26, this is a computer program product with readable medium to embody a program code for performing the same step limitations recited in claim 6; hence is rejected with the corresponding rejection as set forth therein.

As per claims 27-30, these claims correspond to claims 7-10, respectively; hence are rejected with the corresponding rejections set forth therein.

Claim Objections

7. Claims 1, 6, 11, 16, 21, and 26 are objected to because of the following informalities: the limitation recited as 'dynamically-type' must be corrected to become 'dynamically-typed'. Appropriate correction is required.

Response to Arguments

8. Applicant's arguments filed 12/03/2004 have been fully considered but they are not persuasive. Following are the Examiner's observations in regard thereto.

(A) Applicants have submitted that neither OregonU nor APA is directed towards a program language which is dynamically-typed because C++ is known to be a statically typed language (Appl. Rmrks, pg. 7, first 2 paras). The rejection has pointed out how an object can have its type redefined dynamically via C++ type casting. And such definition reads on the newly added limitation. The issue is not whether C++ is a statically or dynamically typed language but whether an object is defined using a dynamically-type language. When a certain type is casted dynamically into a runtime type for a runtime object, then such object entity is defined dynamically, i.e. dynamically typed. The invention as claimed describes a way to retrieve a size or type value; then recites a limitation about dynamic type being defined without providing specifics as to how such dynamic type limitation connects or interrelates to the above size or type

Art Unit: 2193

value retrieving. Omission of steps deemed essential features of an invention such as not providing the claimed features with a clear interrelationship, i.e. features critical or essential to the practice of the invention, but not included in the claim(s) so as to enable clear understanding between the elements recited, is rejectable under USC 112, 2nd paragraph. Notwithstanding such omission, as long as the claim does not explicitly elaborate or give specific description as to what this 'dynamically-typed' limitation amounts to, the limitation as recited is interpreted according to the merits of the language used therefor. As reasonably and broadly interpreted, the C++ casting being integral to C++ language has read into what has been interpreted from the claim.

(B) Applicants have submitted that neither Eckel nor APA is directed towards a program language which is dynamically-typed because C++ is known to be a statically typed language (Appl. Rmrks, pg. 7, bottom, pg. 8 top). This argument falls under the ambit of the arguments being addressed in section A above; and is referred thereto.

For the above reasons, the claims stand rejected as set forth in this Office Action.

Conclusion

9. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37

Art Unit: 2193

CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (272) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

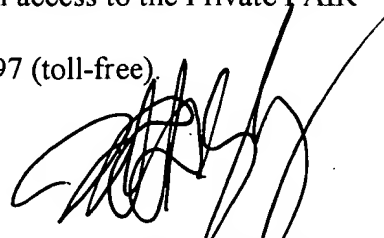
If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571)272-3719.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence – please consult Examiner before using) or 703-872-9306 (for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

VAT
April 28, 2005



TODD INGERBERG
PRIMARY EXAMINER